Pattern Recognition 45 (2012) 2377–2389

Contents lists available at SciVerse ScienceDirect

Pattern Recognition

journal homepage: www.elsevier.com/locate/pr

BLasso for object categorization and retrieval: Towards interpretable visual models

Ahmed Rebai*, Alexis Joly, Nozha Boujemaa

INRIA Rocquencourt, Domaine de Voluceau, Rocquencourt - B.P. 105, Le Chesnay 78153, France

ARTICLE INFO

Article history:
Received 10 November 2010
Received in revised form
26 July 2011
Accepted 26 November 2011
Available online 13 December 2011

Keywords:
Object categorization
Feature selection
Boosting
Lasso
Sparsity
Interpretability

ABSTRACT

We propose a new supervised object retrieval method based on the selection of local visual features learned with the BLasso algorithm. BLasso is a boosting-like procedure that efficiently approximates the Lasso path through backward regularization steps. The advantage compared to a classical boosting strategy is that it produces a sparser selection of visual features. This allows us to improve the efficiency of the retrieval and, as discussed in the paper, it facilitates human visual interpretation of the models generated. We carried out our experiments on the Caltech-*256* dataset with state-of-the-art local visual features. We show that our method outperforms AdaBoost in effectiveness while significantly reducing the model complexity and the prediction time. We discuss the evaluation of the visual models obtained in terms of human interpretability.

1. Introduction

Object recognition in still images has been widely studied in the field of computer vision [1–3]. Tasks may involve classification, retrieval and even detection. Classification, also called *object categorization*, consists in classifying images into categories by giving them the labels of the objects they contain. For each object category, images are attributed the value 1 or 0 depending on whether or not they contain the object. Following the same idea, we can perform object retrieval. Users can query the system to retrieve images which contain the objects they are looking for. This should be based upon an efficient and fast index structure to ensure a reasonable response time, particularly when dealing with large databases and/or complex models. The detection task is trickier because it must provide answers to the two following questions: how many objects are there? Where are they? One may indicate a bounding box in which the object is localized. A way better is to specify a segmented region which defines the boundaries of the object itself.

Object recognition involves many challenges starting with the definition of the training dataset and the choice of the visual descriptors, moving to the way the computer learns and the construction of a reliable classifier. Some researchers prefer to use a bottom-up approach tracing the very low information in the signal and trying to interpret it so as to get powerful models. Others find that it is more intuitive to use a top-down approach [4–6]. They try to characterize the signal by exploiting their knowledge of what the objects are. This raises issues about visual stimulus and how we humans recognize things [7]. For example, is the contextual information always useful? Does it help recognition when scale change occurs or does it make the learning error-prone in the presence of occlusion and clutter? As a rule of thumb, using varied backgrounds during the training improves the generalization ability of the classifier [8]. From now on, we will place ourselves in the context of weak learning. A training image is labeled as a whole sample. It will take the label +1 if it contains the object, −1 if not. This is also known as multiple instance learning. It deals with uncertainty of instance labels. An image is viewed as a bag of multiple features which are the local visual signatures. The bag will have only one label according to whether or not it includes at least one positive instance. It follows that it is only certain for a negative bag that there are no objects. Using a weak learning approach will also help to train images without much knowledge about the objects inside so there will be no need to construct a ground truth per object location.

An object is viewed as a tangible concept. We believe that good recognition comes with a good description, specially one that uses multi-criteria such as shape, texture, scale and color. Image descriptors are indeed the raw material and the basic data for learning. In order to cover the difference in the nature of the objects to be learned and at the same time the intra-class variability of the same object, a multiple description scheme is

* Corresponding author. Tel.: +33 139635196; fax: +33 139635674.
E-mail addresses: ahmed.rebai@hotmail.fr, ahmed.rebai@inria.fr (A. Rebai), alexis.joly@inria.fr (A. Joly), nozha.boujemaa@inria.fr (N. Boujemaa).

0031-3203/$ - see front matter © 2011 Elsevier Ltd. All rights reserved.
doi:10.1016/j.patcog.2011.11.022

needed. It is then up to the learning algorithm to choose a descriptor or a combination of many descriptors that best suits a given category. Interpretability could derive from this fact. Interpretability aims to reduce the semantic gap that exists between human knowledge and the computational representation of the models learned. Computer models are usually too abstract for users to understand where bad results come from. By generating interpretable models, we somehow create a link between the numerical representation of objects and our visual representation. Not only does interpretability enhance our understanding of outputted results but it is also a very effective tool for user interactivity. It allows users to comprehend what the generic model is composed of and to choose, in different situations, the visual patches that best matches their needs. Therefore, interpretability is a means to achieve genericity. Users can perform object retrieval in large database collections which may contain heterogeneous data from different sources.

The paper provides four main contributions. First of all, we define image features that can be easily interpreted in order to help to produce sparse models. Second, we apply the *idea* of the Lasso technique [9] to a multi-instance learning scheme through the use of a modified version of BLasso. The third contribution consists in applying the principle of the Lasso to a discriminative approach for categorical object classification and retrieval. The last contribution is about the models generated. They are interpretable and flexible, thus allowing for user interactivity.

The next section briefly review related works. Then, in Section 3, we describe the algorithm in more detail. After that, we give an overview on the models produced. Section 5 presents the experiments and discusses the results obtained. Finally, we set out our conclusions in Section 6.

2. Related works

This paper proposes a new supervised object retrieval method based on visual local features selection learned with a modified version of the BLasso algorithm. We demonstrate that our method gives equivalent or better prediction performance than AdaBoost while simplifying the object class model. BLasso is an innovative machine learning algorithm which efficiently constrains the loss function. BLasso was introduced by Zhao and Yu [10]. It mixes two successful learning methods: boosting and Lasso. The boosting mechanism was proposed by Schapire [11] in 1990. Since then, many algorithms have emerged [12–16] and boosting has become one of the most successful machine learning techniques. The underlying idea is to combine many weak classifiers – called hypotheses – in order to obtain one final “strong” classifier. Boosting is an additive model which builds up one hypothesis after another by re-weighting the data for the next iteration—increasing the weights of misclassified images and decreasing those of well classified ones. This concept helps to generate different hypotheses, putting emphasis on misclassified examples, typically those located near the decision boundary in the feature space. In addition to that, boosting is able to build a model containing hypotheses of different natures in one learning stage. That is, the feature selection mechanism can process features which belong to different image descriptors. By the term “feature selection”, we mean the process of selecting the most discriminant local signatures of the image.

Boosting has been considered as a stagewise gradient descent method in an empirical cost function, particularly, AdaBoost uses the exponential loss [13,17]. Although it is an intuitive algorithm, boosting may overfit the training data, particularly when it runs for a large number of iterations T in high dimensional and noisy data [17,18]. Moreover, a large value of T implies a long prediction time. On the other hand, setting T to a small value may lead to underfitting. Therefore, the model may be non-discriminant, inconsistent and might not cover the variability inside the category itself. The boosting procedure can also be qualified as oblivious as it always functions in a forward manner aiming to minimize the empirical loss. Although the concept of re-weighting is interesting, at an iteration $t+1$, we have no idea whether the t previous generated hypotheses are good enough or not versus the model complexity.

Tibshirani observed that the ordinary least squares minimization technique is not always satisfactory since the estimates often have a low bias but a large variance. In 1996, he came out with Lasso [9] which shrinks or sets some coefficients to zero. Lasso stands for least absolute shrinkage and selection operator. The idea has two goals: first to gain more interpretation by focussing on relevant predictors and, secondly to improve the prediction accuracy by reducing the variance of the predicted values. Lasso minimizes the L_2 loss function penalized by the L_1 norm on the parameters. This is a quadratic programming problem with linear inequality constraints and it is intractable when the vector of parameters is very large.

In the literature, some efficient methods have been proposed to solve the exact Lasso namely the least angle regression by Efron et al. [19] and the homotopy method by Osborne et al. [20]. These methods were developed specifically to solve the least squares problem (i.e. using L_2 loss). They work well where the number of predictors is small. However, they are not adapted to nonparametric and classification tasks. The advantage of BLasso (Boosted Lasso) lies in its ability to function with an infinite number of predictors and with various loss functions. Unlike the boosting standard, and in order to approximate Lasso solutions, BLasso adds a backward step after each iteration of boosting. Thus, one is able to build up solutions with a coordinate descent manner and then take a look back at the consistency of these solutions regarding the model complexity. It has been demonstrated [10] that BLasso solutions converge to the Lasso path, hence favoring sparsity.

In this paper, we use image features that are easily understandable by humans to help to produce sparse models. Sparsity is preferable because it reduces the model complexity and subsequently the prediction time. Moreover, the features used are mapped to their exact geometric locations in the training images. Therefore, the models generated represent true real entities of what is described and they are not a vague approximation of the image content as it is usually the case with a discriminative training. Our choice here allows the learning algorithm to concentrate on the most useful parts of the object. Furthermore, using a multi-instance approach has the luxury of unsupervised learning where the algorithm tries to find hidden structure and relations between data. It gives indeed more freedom to the algorithm to select background features whenever they turn out to be useful to characterize the category. In addition to that, the models generated are extensible if we ever want to use additional training data. They are also shrinkable and can be modified according to the needs of a human operator. Users can query the retrieval engine using only the visual features that they think are the best for their purpose.

3. Multiple-instance learning with the BLasso mechanism

Boosted Lasso is a machine learning tool which generates sparse models. Producing sparser solutions helps researchers and users to understand what the model is composed of, but this is not guaranteed unless each individual feature contributing to the final model is interpretable. We begin by presenting the image representation chosen.

3.1. Image representation

Most recent and effective recognition techniques [21,22,16] are based on classifiers learned on high-dimensional representations